



TITLE:

# バッチプロセスにおける進捗管理 支援システム

AUTHOR(S):

河野 浩司; 長谷部, 伸治

---

CITATION:

河野 浩司 ...[et al]. バッチプロセスにおける進捗管理支援システム. 化学工学論文集 2008, 34(3): 367-375

ISSUE DATE:

2008

URL:

<http://hdl.handle.net/2433/123418>

RIGHT:

Copyright (c) 2008 (社) 化学工学会; This is not the published version. Please cite only the published version.; この論文は出版社版ではありません。引用の際には出版社版をご確認ご利用ください。

# バッチプロセスにおける進捗管理支援システム

河野 浩司・長谷部 伸治

京都大学大学院工学研究科 化学工学専攻, 615-8510 京都府  
京都市西京区京都大学桂

キーワード：バッチプロセス、進捗管理、スケジューリング、  
運転支援、不確定性

バッチプロセスでは、様々な要因により各工程の所要時間が変動する。所要時間の変動によるスケジュールの遅れは生産性の低下や作業ミスの原因となるため、できるだけ早くもとのスケジュールに戻すことが望ましい。本報では実際に遅延が発生したとき、復旧方法をオンラインでガイダンスするシステムを提案する。提案するシステムでは、まず遅延発生時に取り得る対策を戦略として分類し、戦略ごとに各工程の短縮可能時間と短縮に要するコストを定義した。そして、復旧するまでに許されるバッチ数の制約下で復旧に要するコストを評価として所要時間を短縮する工程とその短縮時間を導出する手法を提案した。提案した最適化手法を組み込んだ遅延回復支援システムのプロトタイプシステムを開発し、その有効性についてシミュレーションにて評価した。

# Progress Control Support System for Batch Processes

Koji KAWANO and Shinji HASEBE

*Department of Chemical Engineering, Kyoto University, Nishikyo-ku,  
Kyoto-shi, Kyoto 615-8510, Japan*

**Keywords:** Batch Process, Progress Control, Scheduling, Operation  
Support System, Uncertainty

In a batch process, the processing time changes according to various unforeseen factors. The delay of pre-assigned schedule lowers productivity and acts as a trigger for operational faults. Thus, the delay should be recovered within a reasonable time period. In the proposed system, several delay recovery strategies are defined in advance, and for each strategy the maximum reduction time and its cost are assigned to each processing stage. Then, the problem of finding the revised schedule is formulated as the problem of optimizing the recovery cost under the constraint of the number of batches within which the schedule should be recovered. A progress control support system using the proposed optimization method has been developed. When the delay of the operation occurs, the proposed system graphically indicates the operations where the processing times should be shortened. The effectiveness of the developed system was evaluated by applying the system to several test problems.

## 結 言

バッチプロセスは多品種少量生産に適しているものの、時間と共にプロセスの状態が変化するため、運転スケジュールを適切に定めないと生産効率の悪いプロセスとなってしまう。しかしながら、様々な不確定要因のため、レシピで規定されている条件や手順で製造しているにもかかわらず、目標品質を得るための所要時間が変動することは避けられない。

所要時間の変動はあらかじめ定めたスケジュールに影響を与え、生産効率を悪化させるばかりでなく、遅延に伴う作業ミスの誘発や納期の遅延を引き起こす。また、物流や検査等他部門のスケジュールにも影響を与える。変動が大きい場合は、変動結果を初期条件として、再スケジューリングが実施され、新たなスケジュールが作成される。しかしながら、すべての変動に対して再スケジューリングを行うのは現実的でなく、小さな変動に対してはバッチ間にあらかじめ設けられたバッファ時間や余裕のある作業の操作時間を短縮してスケジュールの修復が試みられる。従来、このような調整は製造部門において技術者の経験に基づいて行われていた。

著者らは前報（Kawano and Hasebe, 2007）において、バッチプロセスの製造工程には、コストや人手をかければ所要時間を短縮できる工程があることを考慮して、バッチ間に挿入すべきバッファ時間をシステマティックに導出する方法を提案した。提案した手法を用いることにより、あらかじめ定めた範囲の遅延であれば遅れが生じたバッチ、装置、工程に関わらず、あらかじめ定めたバッチ数以内で動的に遅延を吸収できるスケジュール立案が可能である。しかし、提案手法は生産スケジュール立案時にオフライ

ンで利用するための手法であり、運転開始後実際に遅延が発生した場合にどのように対処すべきかについて、具体的な方法は示していなかった。

そこで本報では、前報の手法で立案されたスケジュールで運転されているバッチプロセスにおいて、実際に遅延が発生した場合の具体的な対処方法を運転員に提示する手法を提案する。各バッチでの1つ1つの処理工程を要素とみなせば、その要素の処理順序は技術的に定まっている。よって、各バッチの処理順と処理装置が与えられたとき、上記各要素の処理の順序関係は定まる。処理順序の定まった要素の処理開始時刻の決定問題は、プロジェクトスケジューリング問題として古くから研究が進められている。Sekine, T(1977)は、P A R TやC P Mに関して具体的な計算方法を紹介している。Huercio *et al.* (1995)やFujimura *et al.* (2002)はオンラインでのスケジューリングおよびガントチャート表示機能を提案しているが、そこでは遅延発生分を元に戻すというコントロールの概念は示していない。

これらの点をふまえ本研究では、バッチプロセスを対象にC P Mとリアルタイムガントチャートの考え方を発展させて、遅延発生時の状況に応じた戦略を設定することにより進捗管理に柔軟性を持たせるとともに、オンラインで遅延発生時の復旧方法に関するガイダンスを提示する運転支援システムを提案する。

## 1. 遅延復旧を考慮したスケジュール

### 1.1 対象プロセス

本研究では、説明を簡単にするため直列につながれた $M$ 基のバッチ装置からなるプロセスを対象とする。プロセスが複雑な構造を持つ場合への提案手法の拡張については、5. で説明する。

対象とする製品の 1 バッチの処理全体を 1 つのジョブと呼び、ジョブの各装置での処理順序は与えられていることを前提とする。J 個のジョブを対象とし、各ジョブの装置間での追い抜きはないとする。装置間に中間貯留タンクはなく、移送工程はその次の装置の受入工程と同期する。そして、計画段階では、装置での処理が終了した時点で直ちに次装置へ移送し、装置内で無駄に保持はしないものとする。

## 1.2 工程所要時間の短縮

一般に、原料や助剤の仕込み処理やろ過器のフィルタ交換、洗浄等の作業員が関与する作業は、余裕を持って工程の所要時間が設定されている。言い換えれば、余裕を取らずに作業したり、作業人数を一時的に増員したりすることにより所要時間の短縮が可能である。また加熱や冷却操作は、処理方式やその熱媒もしくは冷媒を変更することにより所要時間の短縮が可能なケースもある。たとえば、インナーコイルとジャケットのどちらかを単独で使用するか併用するか、あるいは再冷水と冷凍水のどちらを利用するかなど、パネル盤のスイッチの切り替えという簡単な操作で工程の所要時間を変えることができ、それにともない処理コストも変化する。

このように、バッチプロセスでは、運転員にわずかな負荷をかけることやコストをかけることにより所要時間を短縮することが可能となるケースがある。本研究では、所要時間の短縮の仕方を、運転員に負荷をかける方法と、ユーティリティ等を用いる方法に分類し、それぞれをコントロール方式と呼ぶことにする。そして、各工程に対してレシピ情報としてコントロール方式ごとに、短縮可能時間および単位時間の短縮に必要なコストの情報が設定されているとする。

1) 運転員対応のコントロール方式：運転員の作業負荷

増による所要時間短縮

2) 用役対応のコントロール方式：利用する用役の変更、

増量による所要時間短縮

本研究では、所要時間の短縮法を上記の2種類のコントロール方式に分類したが、対象プロセスの特徴を考慮して3種類以上に分類しても構わない。いずれのコントロール方式を採用するにせよ、所要時間を変えることができない工程では短縮可能時間およびコストは0とする。運転員対応のコントロール方式を採用した場合、現実には費用は発生しない。しかしながら、運転員への負担増はできるだけ避けるべきであることから、その負担増を数値化し、コストとして表現できるものとした。

### 1.3 遅延復旧を考慮したスケジュール作成法

実際の運転では必ずしもあらかじめ定めたスケジュール(以後、基準スケジュールと呼ぶ)通りに運転が進捗するとは限らない。あるジョブのある工程の所要時間が標準所要時間より延びた場合、そのジョブが基準スケジュールを守れないのはやむをえない。しかしながら、運転員の作業段取りに与える外乱、他の系列とのインターアクションを考えると、できるだけ短期間で基準スケジュールに戻すことが望ましい。このような観点から前報では、運転開始後あらかじめ想定した大きさの所要時間の遅延が発生した際に、あらかじめ定めたバッチ数以内に元の基準スケジュールに戻すことができるスケジュール作成法を提案した。前報で提案した遅延復旧を考慮したスケジュール作成法について、簡単に説明する。

提案手法では、各ジョブについて各装置の各工程での所要時間に対して、考慮すべき最大の遅れ時間がレシピ情報として設定さ

れているものとする．この値は、各工程の所要時間に対して、その変動の標準偏差がわかっているならば、その何倍という形で与えても良いし、経験により設定しても良い．そして、いずれかの工程の所要時間がその最大値だけ延びた場合に、あらかじめ定めたバッチ数以内に遅延を復旧できるスケジュールを可能スケジュールと定義する．遅延復旧のためにはジョブ間にとるバッファ時間の調整と前節で説明したコントロール方式が使えるものとする．ここで、バッファ時間とは、スケジュール作成時にバッチ間に与える時間余裕であり、各装置での2つのジョブ間の遊休時間の中の最小値（ボトルネック工程での遊休時間）である．

各工程の所要時間の短縮に使えるコントロール方式が前節に示した2種類であるとき、可能な戦略としては以下の4通りが考えられる．

- ・ 使用可能なコントロール方式すべてを適用する  
（以後、生産量重視戦略と呼ぶ）
- ・ 運転員対応のコントロール方式のみを適用する  
（以後、運転員対応戦略と呼ぶ）
- ・ 用役対応のコントロール方式のみを適用する  
（以後、用役対応戦略と呼ぶ）
- ・ ジョブ間のバッファ時間のみで対応する  
（以後、バッファ時間利用戦略と呼ぶ）

ジョブ  $j$  のいずれかの装置のいずれかの工程の所要時間がその最大値だけ遅延しても、 $H$ バッチ後のジョブが変動の影響を受けないように制御するためにジョブ間のバッファ時間が満たすべき制約は、以下の手順で求められる．

- 1) ジョブ間に待ち時間をとらないという条件で、最早スケジュールを作成する．この際、1つのジョブを構成する工程間に



においても待ち時間無しで操作する。

- 2) ジョブ  $j$  の各装置の工程に設定した最大遅れ時間に対して、ジョブ  $j$  からジョブ  $j + H$  の間で使える処理時間短縮方法すべてを用いて、できるだけ早く処理するスケジュールを作成する。そして、ジョブ  $j + H$  の最後の装置の開始時刻の、1) で求めた最早スケジュールからの遅れを計算する。ジョブ  $j$  のすべての装置に対して求めたこの遅れの最大値を  $d_j^*$  とする。ただし、この値が負になった場合は、0 とする。
- 3) ジョブ  $j$  のいずれかの工程で最大の遅延が生じた際にジョブ  $j + H$  を遅らさずに運転できる条件は、ジョブ  $j$  からジョブ  $j + H$  の間に  $d_j^*$  以上のバッファ時間を持たせることである。ジョブ  $j$  とジョブ  $j + 1$  の間に設定するバッファ時間を  $b_j$  とすれば、ジョブ 1 からジョブ  $J - H$  のある装置で遅延が生じたとき、 $H$  バッチ後のジョブに影響を与えない条件は以下のようになる。

$$b_j + b_{j+1} + \dots + b_{j+H-1} \geq d_j^* \quad (j = 1, 2, \dots, J-H) \quad (1)$$

$$b_j \geq 0 \quad (j = 1, 2, \dots, J-1) \quad (2)$$

Equations (1)、(2) の不等式を満たす  $J - 1$  個の変数  $b_j$  が存在することは明らかである。上式を満たす中で、ジョブ  $J$  の終了時刻を最も早くする  $b_j$  は、上式を制約として、

$$B = b_1 + b_2 + \dots + b_{J-1} \quad (3)$$

を最小にする線形計画問題を解くことにより求めることができる。前報では、線形計画問題を解かずに、簡単な計算で可能解を導出する手順についても提案した。

## 2. 所要時間短縮工程の決定

## 2.1 問題設定

前章で示した方法でスケジュールを作成することにより、あらかじめ与えた範囲内の変動であれば、あらかじめ与えたバッチ数以内に元のスケジュールに修復できる。しかしながら、前章で示した方法は、実際に遅延が生じた際にどの工程の所要時間を短縮すべきかについては示していない。操作の遅延によりクリティカルパス上の工程は基準スケジュールのそれと変化する可能性がある。また、コントロールによる工程の所要時間の短縮により、クリティカルパス上の工程は動的に変化する。よって、あるジョブのある工程に遅延が発生したとき、もとのスケジュールに戻すためにどのジョブのどの工程の所要時間を短縮すればよいかを経験のみで判断することは困難である。

このような観点から本研究では、運転中に所要時間の変動が発生した際に、最小のコストであらかじめ定めた基準スケジュールに戻す手順を導出する以下の問題を検討する。

### ＜所要時間短縮工程決定問題＞

$J$  個のジョブがあらかじめ定められた基準スケジュールに従って運転されているものとする。また、各工程の所要時間の短縮に使えるコントロール方式も既知とする。このような条件下で、あるジョブ（ジョブ  $L$  とする）のある工程に遅延が発生したとき、あらかじめ定めたジョブ数  $H$  以内に基準スケジュールに戻すような工程の所要時間の短縮方法の中で、コストを最小にする方法を求めよ。ただし、各処理の開始時刻は基準スケジュールの開始時刻よりも早くしてはならず、また、ジョブ  $L$  と  $L + H$  の各工程の所要時間は短縮しないものとする。

## 2.2 所要時間に関する制約条件

上記の問題設定において、ジョブ  $L$  にて遅延が発生して遅くともジョブ  $L + H$  には基準スケジュールに戻すことを想定していることから、ジョブ  $L$  から  $L + H$  の範囲で考えれば良い。説明を簡単にするために各装置での操作は、受入工程、処理工程、払出工程、洗浄工程の4工程からなるものとする。ジョブ  $j$  の装置  $m$  における受入工程の開始時刻を  $t_{j,m,1}$ 、処理工程の開始時刻を  $t_{j,m,2}$ 、払出工程の開始時刻を  $t_{j,m,3}$ 、洗浄工程の開始時刻を  $t_{j,m,4}$  とする。また、レシピで設定されている各工程の標準所要時間を  $P_{j,m,1}$ 、 $P_{j,m,2}$ 、 $P_{j,m,3}$ 、 $P_{j,m,4}$ 、各工程の処理の運転員対応による短縮時間を  $U1_{j,m,1}$ 、 $U1_{j,m,2}$ 、 $U1_{j,m,3}$ 、 $U1_{j,m,4}$ 、そして、用役対応による短縮時間を  $U2_{j,m,1}$ 、 $U2_{j,m,2}$ 、 $U2_{j,m,3}$ 、 $U2_{j,m,4}$  とする。このとき、任意のジョブ  $j$  の装置  $m$  における各工程の開始時刻と工程の所要時間および2つのコントロール方式を適用した際の短縮時間の関係は以下のようになる。

$$t_{j,m,k} + P_{j,m,k} - U1_{j,m,k} - U2_{j,m,k} \leq t_{j,m,k+1} \quad (j=L, L+1, \dots, L+H; m=1, 2, \dots, M; k=1, 2, 3) \quad (4)$$

各装置の払出開始時刻が次の装置の受入開始時刻となる。この関係は以下のように表せる。

$$t_{j,m+1,1} = t_{j,m,3} \quad (j=L, L+1, \dots, L+H; m=1, 2, \dots, M-1) \quad (5)$$

$$U1_{j,m+1,1} = U1_{j,m,3} \quad (j=L, L+1, \dots, L+H; m=1, 2, \dots, M-1) \quad (6)$$

$$U2_{j,m+1,1} = U2_{j,m,3} \quad (j=L, L+1, \dots, L+H; m=1, 2, \dots, M-1) \quad (7)$$

ジョブ  $j$  の装置  $m$  での洗浄工程が終了しなければ、次のジョブの受入工程は開始できない。この関係は次式で表される。

$$t_{j,m,4} + P_{j,m,4} - U1_{j,m,4} - U2_{j,m,4} \leq t_{j+1,m,1}$$

$$(j = L, L+1, \dots, L+H-1; m = 1, 2, \dots, M) \quad (8)$$

各工程における運転員対応による短縮可能時間の最大値および、用役対応による短縮可能時間の最大値  $U1_{j, m, k}^*$ 、 $U2_{j, m, k}^*$  ( $k = 1, 2, 3, 4$ )は、あらかじめレシピに設定されているものとする。よって、1.で説明したさまざまな戦略から1つの戦略を定めることにより、コントロール可能時間の最大値が定まる。すなわち、生産量重視戦略（すべてのコントロール方式を利用）のとき、

$$U1_{j, m, k} \leq U1_{j, m, k}^*, \quad U2_{j, m, k} \leq U2_{j, m, k}^* \quad (9a)$$

用役対応戦略（用役変更でのみ対応）のとき、

$$U1_{j, m, k} = 0, \quad U2_{j, m, k} \leq U2_{j, m, k}^* \quad (9b)$$

運転員対応戦略（運転員の作業変更でのみ対応）のとき、

$$U1_{j, m, k} \leq U1_{j, m, k}^*, \quad U2_{j, m, k} = 0 \quad (9c)$$

バッファ時間利用戦略（コントロールしない）のとき、

$$U1_{j, m, k} = 0, \quad U2_{j, m, k} = 0 \quad (9d)$$

である。また、ジョブ  $L$  およびジョブ  $L+H$  については操作時間の短縮を行わないことから、すべての装置、工程に対して、 $U_{L, m, k}^* = U_{L+H, m, k}^* = 0$  である。

各操作を基準スケジュールでの開始時刻より早く行わないことから、以下の制約が付加される。

$$t_{j, m, k} \geq t_{j, m, k}^* \quad (j = L, L+1, \dots, L+H; m = 1, 2, \dots, M; k = 1, 2, 3, 4) \quad (10)$$

ここで、 $t_{j, m, k}^*$  は、ジョブ  $j$  の装置  $m$  の工程  $k$  の基準スケジュールでの開始時刻である。

ジョブ  $L$  における装置  $m$  の  $s$  番目の工程の標準所要時間が  $d$  だけ遅延した場合、この遅延は以下のようにその工程の開始時刻の遅れとして表現できる。

$$t_{L, m, s} = t_{L, m, s}^* + d \quad (11)$$

### 2.3 評価関数

ジョブ  $j$  の装置  $m$  での各工程の所要時間  $P_{j,m,k}$  を、運転員対応のコントロールにより単位時間短縮した際のコストおよび、用役対応のコントロールにより操作時間を単位時間短縮した際のコスト  $C1_{j,m,k}$ 、 $C2_{j,m,k}$  ( $k = 1, 2, 3, 4$ ) はあらかじめレシピに設定されているものとする。評価関数としては、以下の式を採用した。

$$z = \left\{ \sum_{j=L+1}^{L+H-1} \sum_{m=1}^M \sum_{k=1}^4 (C1_{j,m,k} U1_{j,m,k} + C2_{j,m,k} U2_{j,m,k}) \right\} + \alpha (t_{L+H,M,4} - t_{L+H,M,4}^*) \quad (12)$$

上式右辺第 1 項は短縮のコストを、第 2 項はジョブ  $L+H$  までに基準スケジュールに戻せなかった際のペナルティ項である。 $\alpha$  は重み係数を表し、 $\alpha$  を  $C1_{j,m,k}$ 、 $C2_{j,m,k}$  に比べてかなり大きく設定することにより、遅れを戻すことを優先した評価とすることができる。

以上の説明より、各操作の最適な短縮時間を求める問題は、Eqs. (4)–(11) および変数の非負制約の下、Eq. (12) を最小にする各操作の開始時刻および短縮時間を求める最適化問題となる。

## 3. 支援システム

### 3.1 変動時の対応

前章において最適な処理開始時刻を求める問題が、線形計画問題として定式化できることを示した。現実のプラントでは、どのジョブのどの工程で遅延が発生するかわからない。本研究で提案するシステムはオンラインでの利用を前提としていることから、遅れが発生した際に効率よく最適化問題を計算機内に構築し解く必要がある。そこで本章では、オンラインでの運用を想定して開発した遅延回復支援システムのプロトタイプシステムの概要を説

明する．

シーケンス制御により自動運転されているバッチプロセスであれば、装置別にジョブ開始、ジョブ終了、および各工程の終了を通知する信号（イベント）をシーケンスに定義し、かつ、装置別に現在運転中のジョブ No とレシピ No を記憶しておくことにより、以下の処理の流れにて運転支援システムが構築できる．

- ① 一定周期（例えば 1 分）でイベントを監視
- ② イベントの発生を検知した際に、発生装置、発生ジョブ No、該当レシピ No、イベントの内容を把握
- ③ イベント発生時刻と立案されたスケジュール上の時刻と比較し、遅延の有無を確認
- ④ 遅延があれば、対策を取るか否かを判断（運転員の判断）．

対策を施さない場合は①へ戻る．また、再スケジュールリングが必要と判断した場合は、スケジュールリングシステムに情報を伝達する．製造部門で対応すべきと判断された場合は、次の⑤に進む．

- ⑤ 遅延回復支援システムに対して必要情報を送付する．
- ⑥ 遅延回復支援システムが与えられた戦略、コスト等の情報に基づいて各ジョブの最適な開始時刻を算出する．また、運転員の要求により遅延回復のためのガイダンス情報を提供する

### 3.2 入力情報

前節の⑤のステップで必要とされる情報について示す．

ジョブ  $j$  に関する変数ベクトル  $\mathbf{X}_j$  を次のように定義する．

$$\mathbf{X}_j = [\mathbf{x}_{j1} \ \mathbf{x}_{j2} \ \cdots \ \mathbf{x}_{jM}] \quad (13)$$

ただし、 $\mathbf{x}_{jm}$  はジョブ  $j$  の装置  $m$  に関する以下の変数ベクトルである．

$$\mathbf{x}_{jm} = [t_{jm1} \ t_{jm2} \ t_{jm3} \ t_{jm4} \ U1_{jm1} \ U2_{jm1} \ U1_{jm2} \ U2_{jm2} \ U1_{jm3} \ U2_{jm3} \ U1_{jm4}]$$

$$U2_{jm4}] \quad (14)$$

この  $\mathbf{X}_j$  を用いれば、Eq. (4) の関係は Eq. (15) のように、Eqs. (5) から (7) の関係は Eq. (16) のように、また Eqs. (9)、(10) の関係はそれぞれ Eqs. (17)、(18) のように表すことができる。

$$\mathbf{A}_1 \mathbf{X}_j^T \geq \mathbf{B}_{1j}^T \quad (j = 1, 2, \dots) \quad (15)$$

$$\mathbf{A}_2 \mathbf{X}_j^T = 0 \quad (j = 1, 2, \dots) \quad (16)$$

$$\mathbf{A}_3 \mathbf{X}_j^T \geq \mathbf{B}_{3j}^T \quad (j = 1, 2, \dots) \quad (17)$$

$$\mathbf{A}_4 \mathbf{X}_j^T \geq \mathbf{B}_{4j}^T \quad (j = 1, 2, \dots) \quad (18)$$

ここで、 $\mathbf{A}_1$ 、 $\mathbf{A}_2$ 、 $\mathbf{A}_3$ 、 $\mathbf{A}_4$  は各式の係数からなる  $j$  に依存しない行列である。 $\mathbf{B}_{1j}$  はジョブが定まればレシピ情報から値がすべて定まる定数ベクトル、 $\mathbf{B}_{3j}$  は用いる戦略が定まればレシピ情報から値がすべて定まる定数ベクトル、 $\mathbf{B}_{4j}$  は基準スケジュールでの開始時刻からなる定数ベクトルである。また、上付きの  $T$  は転置を意味する。

2 つのジョブの変数間の関係である Eq. (8) は、以下のように表現できる。

$$\mathbf{A}_5 [\mathbf{X}_j \mathbf{X}_{j+1}]^T \geq \mathbf{B}_{5j}^T \quad (j = 1, 2, \dots) \quad (19)$$

ここで  $\mathbf{A}_5$  は  $j$  に依存しない行列であり、また  $\mathbf{B}_{5j}$  は装置  $j$  の標準洗浄時間からなる定数ベクトルである。

ジョブ  $L$  の遅延に関する制約式 Eq. (11) は次式で表現できる。

$$\mathbf{A}_6 \mathbf{X}_L^T = \mathbf{B}_6 \quad (20)$$

ここで  $\mathbf{A}_6$  は遅延発生工程を特定するための横ベクトル、 $\mathbf{B}_6$  は遅延の大きさに依存する定数である。

注目すべき点は、 $\mathbf{A}_1$  から  $\mathbf{A}_5$  の構造がジョブに依存しないことである。よって、係数行列をあらかじめ遅延回復支援システムに一度与えておけば、遅延発生時に同システムに与えるべき情報は、i) 遅延発生ジョブの番号、ii) 遅延発生装置および発生工程の番号、

iii)遅延時間、iv)基準スケジュールに戻したいジョブ数  $H$ 、v)コントロール戦略、に限定できる。

これらの情報を与えることで、Eqs. (15)から(20)の右辺の定数ベクトルおよび、Eq. (20)の  $\mathbf{A}_6$  の情報、および評価関数の係数ベクトルの情報が定まる。よって、上記の情報を遅延回復支援システムに送ることにより、遅延発生状況下での最適な所要時間短縮法を算出することができる。

このように本研究の方式では、遅延回復支援システムとのインタフェース情報が非常にシンプルであるため、オンラインでの運用に適しており現実的な方式であると考えられる。今回、線形計画問題を解くためのソルバーとして Windows XP の環境で実行可能なフリーソフトの Lp\_solve 5.5.0.10 を採用した。必要な情報の入力とそのソルバーへの伝達、および計算結果をガイダンスとして表示するインタフェースは、マイクロソフト社の MS-EXCEL にて作成した。インタフェースの画面表示例を **Figure 1** に示す。画面の詳細については次章で説明する。

実際に製造現場にて利用されているバッチコントロールシステムに本研究で提案した運転支援機能を付加した構成図を **Figure 2** に示す。図中の破線部が開発した遅延回復支援システムである。レシピ管理システムは、工程進捗管理システムから得られたスケジュールに基づいてレシピ情報を取り込み、コントローラに伝達する。コントローラはその情報を元に各バッチを実行し、プロセスから発生するイベント信号をイベント管理システムに送付する。工程進捗に関わる情報は、イベント管理システムから工程進捗管理システムに送られ、操作の進捗状況がガントチャートで表示される。遅延が発生したとき、その情報は今回開発した遅延回復支援システムへ送付され、その状況がガントチャートに表示される。



運転管理者は遅延回復支援システムにより定めた戦略の下での遅延回復スケジュールを確認する．そしてその情報を元に、コントロールを実施するか、または再スケジュールリングを実施するかを判断する．

## 4．シミュレーションによる検証

### 4.1 対象プロセス

本研究で提案した遅延回復支援システムの有効性をシミュレーションにて検証した．本章では **Figure 3** の Type 1 のプロセスを対象とした．このプロセスは、Unit 1 から Unit 5 の 5 基のバッチ装置から構成され、すべてのジョブはこの番号順に装置を使用する．各ジョブの各装置での操作は、受入、処理、移送、洗浄の 4 工程からなる．

スケジュール対象の製品は A、B、C の 3 種類であり、そのレシピを **Table 1** に示す．各装置の各工程で予想される最大遅延時間を Max. Delay Time 欄に、コントロール可能時間および単位時間の短縮に必要なコストを Controllable Time & Cost Data 欄に示した．コントロール方式としては、運転員の作業変更および用役変更の 2 種類とした．

### 4.2 シミュレーション結果

対象プロセスにおいて、生産量重視戦略のもとで作成された基準スケジュールに対して遅延が発生したケースを例に検証する．

ジョブ 1 からジョブ 6 が、製品 A、C、C、B、A、A である製造スケジュールに対して、ジョブ 1 の装置 3 の処理工程で最大の遅延が発生したケースを検証した．基準スケジュールに戻したいジョブ数  $H$  を 3 とし、生産量重視戦略をとるものとした．

本ケースのガイダンス画面を **Figure 1** に示す．図中の

Influence of Delay 欄に、遅延が発生したジョブ、装置、工程の番号が表示される。また、基準スケジュールに戻すのに必要なコントロールの内容が Progress Control Guidance 欄に運転員対応のコントロール方式 (Work Change) と用役対応のコントロール方式 (UTT Change) に分けて示される。そして、図中の左下欄にその際のコスト情報が表示される。Optimized Gantt Chart Display ボタンを押すことにより、**Figure 4** のような遅延発生ジョブ以降  $H$  バッチ分のガントチャートが表示される。

ガントチャートは、上段と下段に分かれており、上段にコントロールを実施しない場合 (Non-Controlled) を表し、下段にガイダンス通りコントロールを実施した場合 (Controlled) を表示する。そして、それぞれのガントチャートはユニット毎に基準スケジュール (Initial) と遅延発生後のスケジュール (Delay) を上下に対比させて表示する。ここで斜線が施された矩形は、遅延が発生したジョブおよび装置を表す。また、図中破線で縁取りされた矩形は、ガイダンスで示されているコントロール対象のジョブおよび装置であり、その中の濃く塗りつぶされた部分はコントロール対象の工程を表している。

ガイダンス画面の Strategy を変更することにより、戦略を変えられる。1 例として、生産量重視から運転員に負荷を与えない用役対応に変更した場合のガントチャートを **Figure 5** に示す。戦略変更の結果、用役対応のコントロールのみとなり遅延を吸収できなくなっている。そこで、ガイダンス画面の Strategic Schedule 欄の基準スケジュールに戻したいジョブ数  $H$  (Recovery Job) を 4 に変更した結果、**Figure 6** に示すように 4 ジョブ目では遅延は吸収された。

遅延が生じた際、あらかじめ定めたジョブ数  $H$  以内では遅れを

吸収できないケースがある．その際は遅延を吸収できるまで  $H$  を増やし、その期間が戦略上許容できるものであれば、そのガイドランスに従えばよい．許容できない場合は戦略を変更するか、あるいはスケジュールの見直しを実施する等の対応が必要となる．そして、提案したシステムは、そのようなさまざまな対応を検討するツールとしても有効である．

以上、例題により、従来運転員に委ねられていた進捗コントロールに対して、戦略に応じたガイドランスをシステムティックに提供することが可能であることがわかり、提案した手法の有効性が確認できた．

## 5．複雑なプロセスへの拡張

前章までは直列につながれたバッチ装置からなるプロセスを対象に論じてきた．本章では、2系列が共有装置を有する場合や中間製品の混合を含む場合など、複雑な構造を有するプロセスについても、本論文で提案した手法が適用できることを示す．

### (1) X 型プロセス

Figure 3 の Type 2 のように、2つの系列が特定の装置を共有するケースについて考える．ジョブ  $j$  の装置  $m$  における第  $k$  工程の開始時刻  $t_{j, m, k}$  の系列  $i$  での値を、上付き添え字を用いて  $t_{j, m, k}^i$  と表す．1つの系列内でのジョブの各工程の開始時刻に関する制約は、1系列の場合と同じであることから、Eqs. (4) から (8) がそのまま利用できる．共有装置では2つの系列に属するジョブの関係を設定する必要がある．

共有装置でのジョブの処理順が与えられれば、その装置でのジョブ間の処理開始時刻に関する制約は、Eq. (8) と同様の以下の式

で表すことができる．

$$t_{j(n),m(n),4}^{i(n)} + P_{j(n),m(n),4}^{i(n)} - U1_{j(n),m(n),4}^{i(n)} - U2_{j(n),m(n),4}^{i(n)} \leq t_{j(n+1),m(n+1),1}^{i(n+1)} \quad (21)$$

ここで、左辺各変数の添え字は共有装置で  $n$  番目に処理されるジョブに対するもので、 $i(n)$  は属する系列番号、 $j(n)$  はジョブ番号、 $m(n)$  は装置番号を表す．

たとえば、共有装置で 8 番目に処理されるジョブが系列 1 のジョブ 5、9 番目に処理されるジョブが系列 2 のジョブ 4 であり、共有装置は系列 1 では装置 3、系列 2 では装置 2 である場合、この 2 つの処理の順序関係は以下のように表すことができる．

$$t_{5,3,4}^1 + P_{5,3,4}^1 - U1_{5,3,4}^1 - U2_{5,3,4}^1 \leq t_{4,2,1}^2 \quad (22)$$

よって、共有装置におけるジョブの処理順から得られる  $\{i(n), j(n), m(n) : n=1, 2, \dots, J; J \text{ はジョブ数}\}$  の情報に基づいて、Eq. (22) を新たに制約に加えることにより、1 系列の場合と同様に考えることができる．評価関数については、各系列について Eq. (12) で得られる  $z$  の和を用いればよい．

## (2) Y 型プロセス

Figure 3 の Type 3 のように中間製品が途中の装置で混合されるケースについて考える．通常このようなケースでは、混合される一方が従属的な装置群（以後、副ラインと呼ぶ）であり、その装置群に属する装置の稼働時間には余裕がある場合が多い．このような場合は、副ラインを除いた 1 系列プロセスと考えればよい．副ラインを同時に考慮する必要がある場合は、以下のように考えることができる．

説明を簡単にするため、4. までは 1 つの装置での操作は、受入、処理、払出、洗浄の 4 工程からなるものとした．ここでは一般化し、各装置での操作は  $K$  工程からなるものとする．装置によって必要な工程数が異なる場合はその最大値を  $K$  とし、工程数の

少ない装置には処理時間 0 の工程を追加する．このように考えると、装置間の流れの結合関係は、(払出装置番号, 払出工程番号, 受入装置番号, 受入工程番号) の組で表すことができる．すべての装置での流れの結合関係からなる集合を  $S$  とすれば、装置間の払出と受入の制約条件である Eqs. (5) から (7) の関係は、以下の式で表現できる．

$$t_{j,m,k} = t_{j,m',k'} \quad (23)$$

$$U1_{j,m,k} = U1_{j,m',k'} \quad (24)$$

$$U2_{j,m,k} = U2_{j,m',k'} \quad (25)$$

ここで、 $(m, k, m', k') \in S, j=1, 2, \dots, J$  である．

例えば、装置 5 の第 1 単位操作が装置 2 の第 3 単位操作からの受入で、装置 5 の第 3 単位操作が装置 4 の第 3 単位操作からの受入であれば、 $S$  は  $(5, 1, 2, 3)$ 、 $(5, 3, 4, 3)$  という要素を含み、開始時刻の制約条件は次式となる．

$$t_{j,2,3} = t_{j,5,1} \quad (26)$$

$$t_{j,4,3} = t_{j,5,3} \quad (27)$$

このように装置間の入出力関係を一般化しても、各装置での関係式は変化しないことから、入力情報としては 3.2 の Eq. (16) の係数行列が変化するのみで、他の入力情報は変化しない．よって、上述した部分を修正するのみで前章までの手法を適用できる．

## 結 言

あらかじめ定めた範囲の操作の遅れに対して、各工程の所要時間が短縮可能という条件下で、あらかじめ定めたバッチ数以内で遅延を吸収するために、バッチ間に挿入すべきパuffer時間をシステマティックに導出する方法を前報において提案した．本報で

は実際に遅延が発生したとき、遅延発生状況に応じて柔軟に戦略を指定し、それに基づいた復旧方法をオンラインでガイダンスするシステムを提案した。提案したシステムでは、遅延発生時に少数の情報を入力するのみで、与えた戦略に基づいてコストが最小となる復旧法を提示する。これは、これまで運転員が状況に応じて処理時間を短縮する工程を決めていた考え方に近いものであり、実プロセスに受け入れやすい考え方である。

バッチプロセスに関して、オフラインのスケジューリングの分野では最適化等の様々な研究が報告されてきているが、進捗コントロールの最適化に関する検討は現時点では見当たらない。これまで経験に基づいて行われてきたスケジュール遅延時の対応に最適化の概念を導入することにより、遅延時の対応の標準化とコスト削減が図れ、プロセスの安定運転にもつながると期待できる。

本研究では、変動が生じることを前提に議論してきたが、変動を抑制することも重要である。今後、進捗変動要因の排除という観点から不確定な情報を減少させるために、製品品質を作り込むメカニズムの解明、製造条件のレシピからのずれが操作時間に与える影響の明確化、製品品質を決定するプロセス状態量の明確化等に関する研究を進めていく必要がある。

## Nomenclature

$A_j$	= constraint matrix in Eqs. (15) to (20)	[-]
$B_{ij}$	= constraint vector in Eqs. (15) to (20)	[-]
$b_j$	= buffer time between job $j$ and job $j+1$ at bottleneck unit	[min]
$C1_{j,m,k}$	= cost of unit time reduction by work change	[kYen/min]
$C2_{j,m,k}$	= cost unit time reduction by utility change	[kYen/min]
$d$	= delay	[min]
$d_j^*$	= maximum delay of job $j+H$ when the disturbance occurs at job $j$	[min]
$H$	= number of batches in which the disturbance is absorbed	[-]
$J$	= number of jobs	[-]
$K$	= number of operations	[-]
$M$	= number of batch units	[-]
$P_{j,m,q}, P_{j,m,q}^i$	= processing time	[min]
$S$	= set which shows the connection of flows	[-]
$t_{j,m,k}^*$	= start time determined by scheduling system	[min]
$t_{j,m,k}, t_{j,m,k}^i$	= start time	[min]
$U1_{j,m,k}, U1_{j,m,k}^i$	= control time by work change	[min]
$U1_{j,m,k}^*$	= maximum control time by work change	[min]
$U2_{j,m,k}, U2_{j,m,k}^i$	= control time by utility change	[min]
$U2_{j,m,k}^*$	= maximum control time by utility change	[min]
$x_{j,m}$	= vector consisting of variables related to unit $m$ for	

	job $j$	[-]
$\mathbf{x}_j$	= vector consisting of variables related to job $j$	[-]
$\alpha$	= penalty cost for the delay	[kYen]
$r$	= turn of using shared unit suffix	[-]
$i(n)$	= production line number	[-]
$j, j(n)$	= job number	[-]
$k$	= operation number: receiving ( $k=1$ ), processing ( $k=2$ ), discharging ( $k=3$ ), and washing ( $k=4$ )	[-]
$m, m(n)$	= batch unit number	[-]



## Literature Cited

- Kawano, K. and S. Hasebe; ” A Method for Setting Buffer  
Times between Batches with Uncertain Processing  
Times “, *Kagaku Kougaku Ronbunshu*, **33**, 354-362 (2007)
- Sekine, T.; PERT・CPM OR Library 11, 2nd ed., pp. 208-225,  
Nikkagiren Shuppan-sha, Japan (1977)
- Huercio, A., A. Espuna and L. Puigjaner; ” INCORPORATING  
ON-LINE SCHEDULING STRATEGIES IN INTEGRATED BATCH  
PRODUCTION CONTROL” , *Computers Chem. Engng* Vol.19,  
Suppl., pp. S609-S614 (1995)
- Fujimura, S. and R. Himono; ” Unified Recipe Information  
Management for Batch Process Control and Scheduling” ,  
*T. IEE Japan*, Vol. 122-C, No.5, pp. 843-850 (2002)

図 表

**Operation Support System**

**Production Planning** ProductA 1 ProductC 2 ProductB 1 ProductA 2 **Close**

Start Time (Sum of minute expression) 0

**Earliest Schedule** **Earliest Schedule Generate**

**Strategic Schedule** Recovery Job → 3 Jobs after **Strategic Schedule (Non Controlled) Generate**

**Strategy**  
☒ Maximum Production ☐ UTT Change Control ☐ Work Change Control

**Strategic Schedule (Controlled) Generate**

**Influence of Delay** Delay Occurred → Job No 1 Unit No 3 Operation No 2 **Prediction of Influence** **Delay Influence Gantt Chart**

**Optimization**

**Constraint Generate**

**Guidance Display**

Work Change Cost [KYen] 11.0  
 UTT Change Cost [KYen] 38.0  
 Delay Penalty Cost [KYen] 0.0  
 Total Cost [KYen] 49.0

**Optimized Gantt Chart Display**

**Progress Control Guidance**

Job No: 2	Unit No: 3	Target: Processing	Control Time: 10min
Job No: 2	Unit No: 3	Target: Discharging	Control Time: 3min
Job No: 2	Unit No: 4	Target: Receiving	Control Time: 3min
Job No: 3	Unit No: 1	Target: Discharging	Control Time: 4min
Job No: 3	Unit No: 2	Target: Receiving	Control Time: 4min
Job No: 3	Unit No: 2	Target: Processing	Control Time: 5min

Job No: 2	Unit No: 3	Target: Processing	Control Time: 17min
Job No: 3	Unit No: 2	Target: Processing	Control Time: 21min

**Work Change**

**UTT Change**

Fig. 1 Control guidance under maximum production strategy

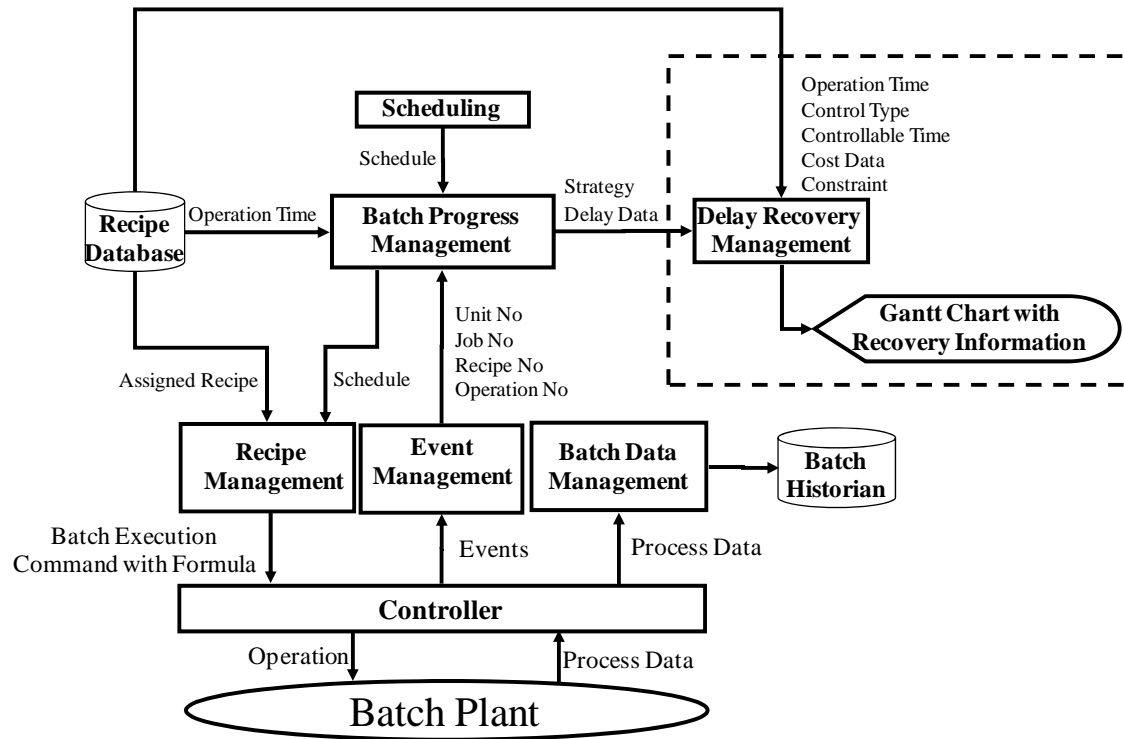
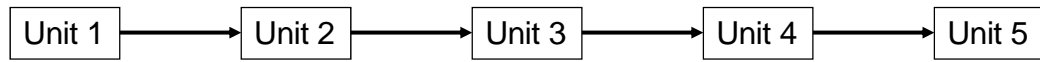


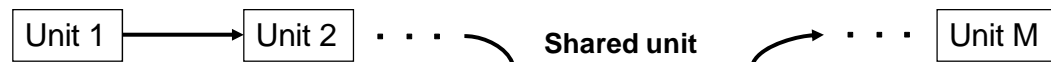
Fig. 2 Batch control system

Type 1



Type 2

Line 1



Line 2



Type 3

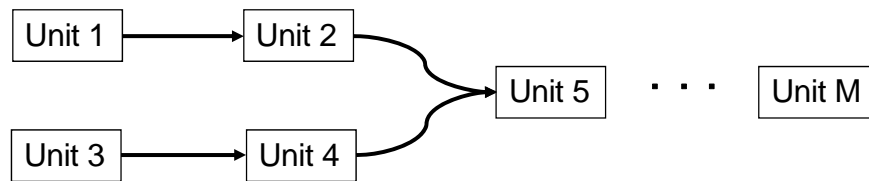


Fig. 3 Target process

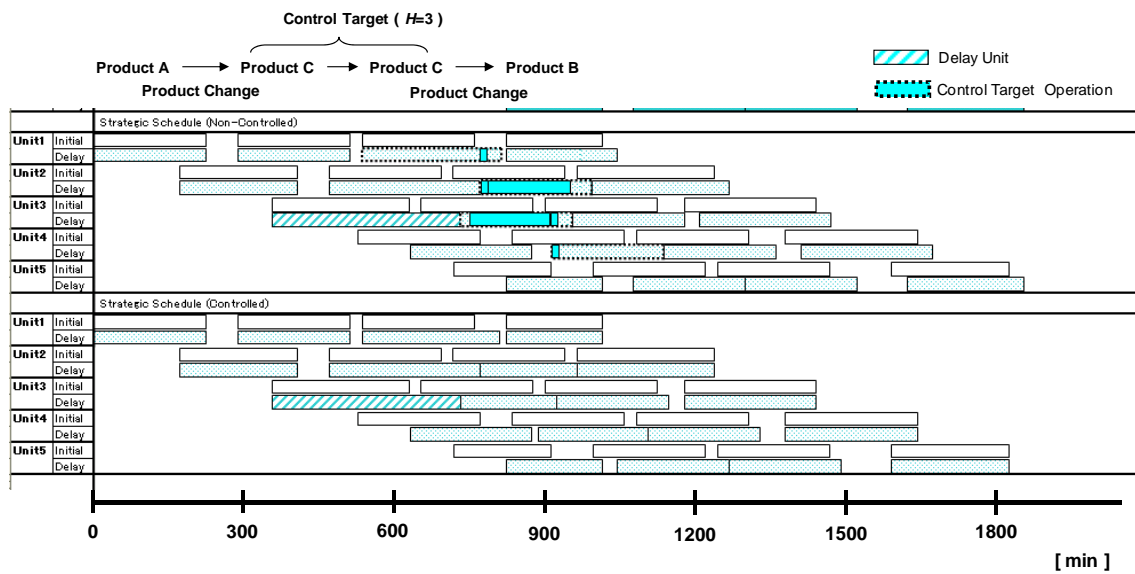


Fig. 4 Optimized Gantt chart under maximum production strategy

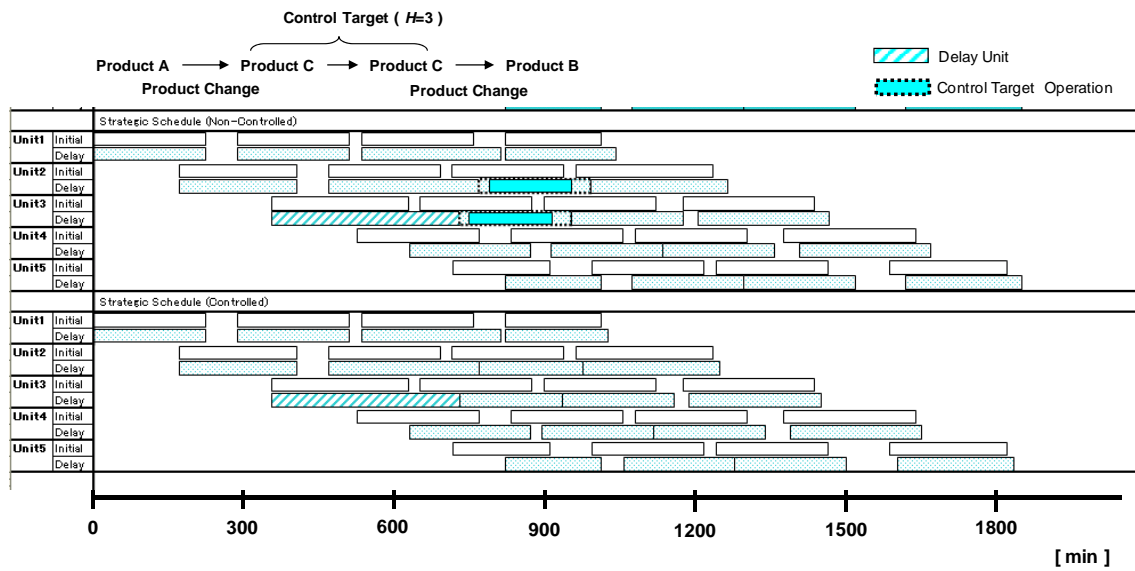


Fig. 5 Optimized Gantt chart under utility change strategy  
(  $H = 3$  )

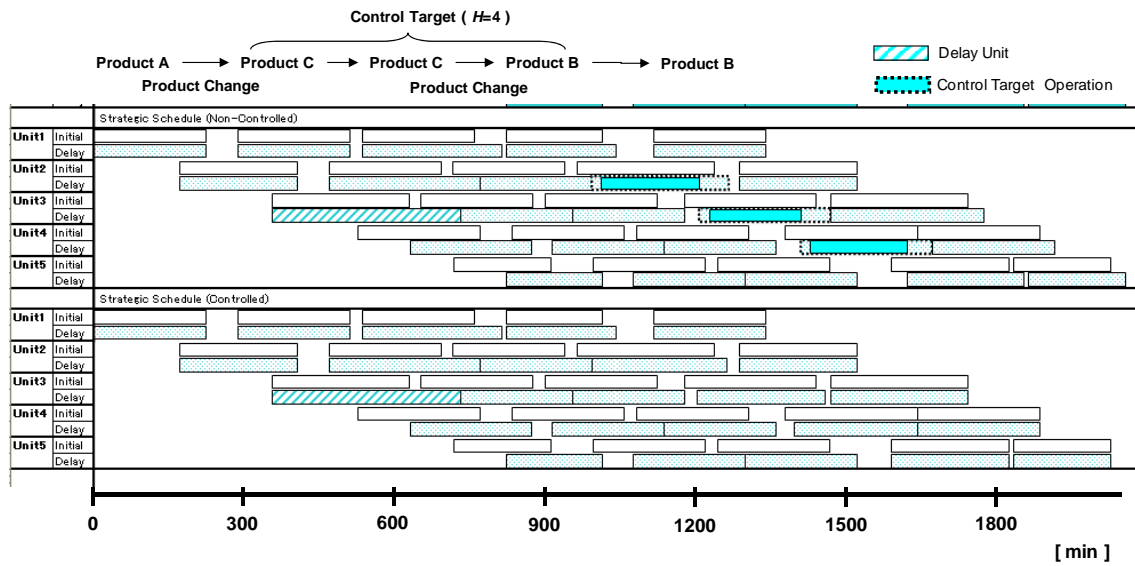


Fig. 6 Optimized Gantt chart under utility change strategy  
( $H = 4$ )

Table 1 Target recipe

Time : [min] , Cost : [kYen/min]																					
Product A								Product B						Product C							
Unit No		Operation Type		Oper- ation Time	Max. Delay Time	Controllable Time & Cost Data				Oper- ation Time	Max. Delay Time	Controllable Time & Cost Data				Oper- ation Time	Max. Delay Time	Controllable Time & Cost Data			
						Work Change		UTT Change				Work Change		UTT Change				Work Change		UTT Change	
						Time	Cost	Time	Cost			Time	Cost	Time	Cost			Time	Cost	Time	Cost
1	RECEIVING	20	0	0	0	0	0	20	0	0	0	0	0	20	0	0	0	0	0		
	PROCESSING	150	69	0	0	3	1.00	120	69	6	0.50	5	1.00	160	51	13	0.50	11	1.00		
	DISCHARGING	20	0	5	0.50	0	0	20	0	0	0	0	0	20	0	4	0.50	0	0		
	WASHING	30	0	0	0	5	0.50	30	0	0	0	10	0.50	20	0	0	0	0	0		
2	RECEIVING	20	0	5	0	0	0	20	0	0	0	0	0	20	0	4	0.25	0	0		
	PROCESSING	160	99	10	0.50	5	1.00	190	90	12	0.50	4	1.00	160	33	5	0.50	28	1.00		
	DISCHARGING	20	0	0	0	0	0	20	0	5	0.50	0	0	20	0	0	0	0	0		
	WASHING	30	0	0	0	10	0.50	40	0	0	0	12	0.50	20	0	0	0	0	0		
3	RECEIVING	20	0	0	0	0	0	20	0	5	0.25	0	0	20	0	0	0	0	0		
	PROCESSING	170	102	10	0.50	5	1.00	180	117	6	0.50	8	1.00	160	27	10	0.50	17	1.00		
	DISCHARGING	20	0	0	0	0	0	20	0	0	0	0	0	20	0	3	0.50	0	0		
	WASHING	80	0	0	0	20	0.50	40	0	0	0	10	0.50	20	0	0	0	0	0		
4	RECEIVING	20	0	0	0	0	0	20	0	0	0	0	0	20	0	3	0.25	0	0		
	PROCESSING	170	96	8	0.50	12	1.00	190	90	13	0.50	18	1.00	140	33	8	0.50	16	1.00		
	DISCHARGING	20	0	0	0	0	0	20	0	0	0	0	0	20	0	0	0	0	0		
	WASHING	30	0	0	0	6	0.50	30	0	0	0	6	0.50	40	0	0	0	10	0.50		
5	RECEIVING	20	0	0	0	0	0	20	0	0	0	0	0	20	0	0	0	0	0		
	PROCESSING	90	27	6	0.50	0	0	150	108	10	0.50	15	1.00	160	27	8	0.50	17	1.00		
	DISCHARGING	20	0	0	0	0	0	20	0	0	0	0	0	20	0	0	0	0	0		
	WASHING	60	0	0	0	14	0.50	40	0	0	0	5	0.50	20	0	0	0	0	0		